Théorie des graphes

L3 TECHNOLOGIES DE L'INFORMATION

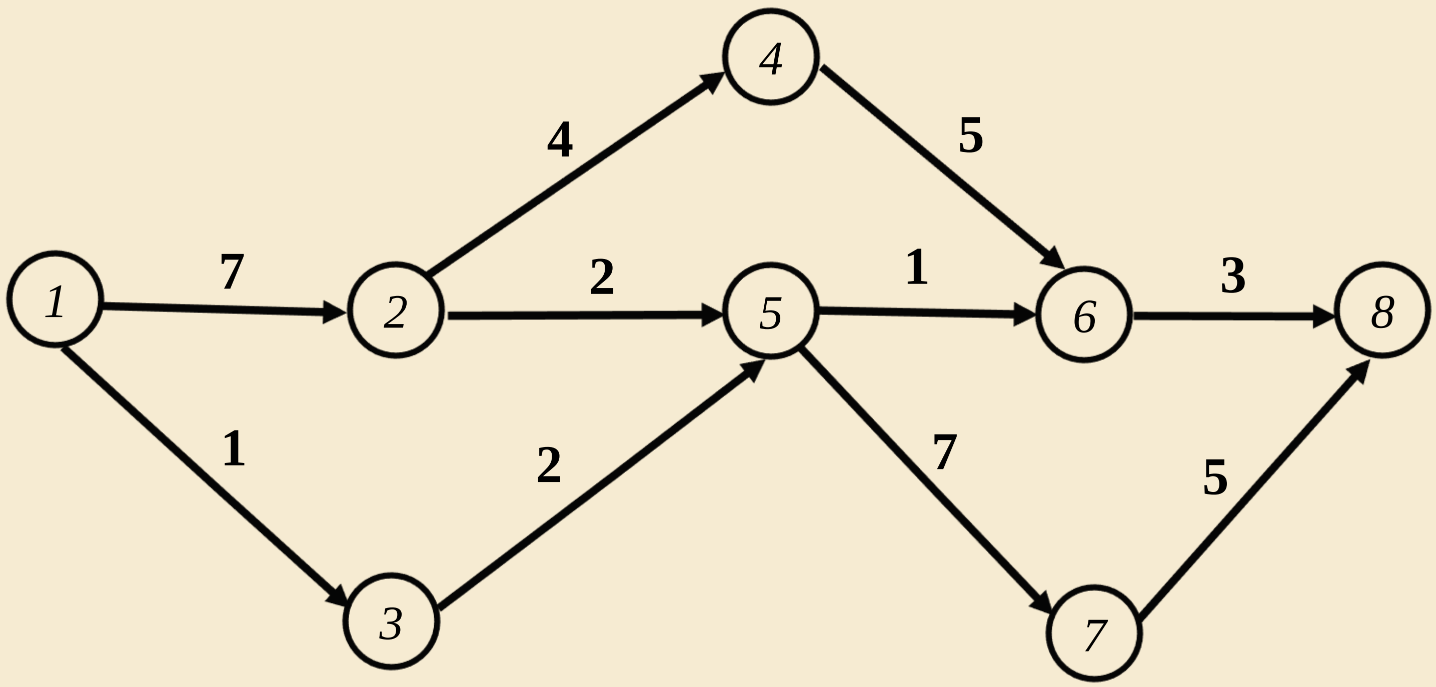
DR ABDALLAH EL KHYARI
ABDALLAH.ELKHYARI@GMAIL.COM

INTRODUCTION

Analyse des problèmes (1)

Recherche de chemin

- **Un ou des chemin(s) reliant les sommets 1 à 8**



Analyse des problèmes (2)

Problème ExisteChemin :

- Instance : un graphe orienté et valué G, et 2 sommets 1 et 8.
- Question : existe-t-il un chemin dans G qui relie le sommet 1 à 8 ?

Problème TrouveChemin :

- Instance : un graphe orienté et valué G, et 2 sommets 1 et 8.
- Question : trouver un chemin dans G reliant le sommet 1 au sommet 8.

Problème TousLesChemins :

- Instance : un graphe orienté et valué G, et 2 sommets 1 et 8.
- Question : trouver tous les chemins dans G qui relient le sommet 1 à 8.

Problème PlusCourtChemin :

- Instance : un graphe orienté et valué G, et 2 sommets 1 et 8.
- Question : trouver un plus court chemin dans G reliant le sommet 1 à 8.

Analyse des problèmes (3)

Problème de décision

- C'est un problème qui prend en entrée une instance, et répond en sortie "oui" ou "non" suivant si l'instance répond au problème ou pas.

Problème d'existence (recherche d'une solution)

- C'est un problème qui comporte une question ou plutôt une injonction de la forme « *trouver un élément tel que …* » dont la réponse consiste à fournir un tel élément.

Problème d'énumération (recherche de toutes les solutions)

- C'est un problème qui comporte une question ou plutôt une injonction de la forme « *trouver tous les éléments tel que …* » dont la réponse consiste à fournir un ensemble d'éléments.

Problème d'optimisation (recherche de la meilleure solution)

- C'est un problème d'existence qui consiste à fournir une solution qui optimise un certain critère (coût, profit, …).

Problème décision/optimisation

Problèmes de décision

- Réponse = oui ou non
- Exemple : Peut-on colorier ce graphe avec 4 couleurs ?

Problèmes d'optimisation

- Réponse = valeur qui optimise une fonction objectif donnée
- Exemple : Plus petit nombre de couleurs permettant de colorier ce graphe ?

Problème de décision vs problème d'optimisation

- Existe t'il une meilleure solution ?
- Exemple : Peut-on colorier ce graphe avec moins de 5 couleurs ?

Remarque

- L'équivalence entre ces deux problèmes suppose que la démonstration d'existence repose sur un argument constructif

Problèmes & algorithmes

Définitions

- Un **problème de décision** est dit **décidable** s'il existe un algorithme, une procédure mécanique qui termine en un nombre fini d'étapes, qui le décide, c'est-à-dire qui répond par oui ou par non à la question posée par le problème

- S'il n'existe pas de tels algorithmes, le problème est dit **indécidable**

Remarques

- Un problème **décidable** peut admettre plusieurs algorithmes qui le résolvent
- Exemple : recherche d’un élément dans une liste

Complexité algorithmique (1)

Définition

- La complexité d'un algorithme A est une fonction $C_A(N)$, donnant le nombre d'instructions caractéristiques exécutées par A dans le pire des cas, pour une données de taille N.

Pire cas

- pour N fixé, on considère la donnée donnant le plus de travail à l'algorithme. Une complexité moyenne serait utile mais difficile à calculer.
- La complexité en moyenne nécessite une connaissance de la distribution probabiliste des données.

Complexité algorithmique (2)

Algorithme de tri par sélection

- Le pseudo-code de l'algorithme s'écrit :

```
procédure tri_selection(tableau t, entier n)
    pour i de 1 à n - 1
        min = i
        pour j de i + 1 à n
            si t[j] < t[min] alors min = j
        fin pour
        si min ≠ i alors échanger t[i] et t[min]
    fin pour
fin procédure
```

- Complexité de l'algorithme : **O(n²)**

Complexité algorithmique (3)

Trier un tableau de n éléments

- Algorithme de tri par sélection : $\mathbf{O(n^2)}$
- Algorithme de tri par insertion :
  - dans le meilleur des cas (tableau déjà trié) : $\mathbf{O(n)}$
  - dans le pire des cas (tableau trié à l'envers) : $\mathbf{O(n^2)}$
- Algorithme de tri rapide (quicksort) :
  - dans le meilleur des cas (pivot = médiane) : $\mathbf{O(n.log(n))}$
  - dans le pire des cas (pivot = elt max ou min) : $\mathbf{O(n^2)}$
- Algorithme de tri par tas : $\mathbf{O(n.log(n))}$

Classification des problèmes (1)

La classe P (Deterministic Polynomial time)

- L'ensemble des **problèmes de décision** qui peuvent être résolus par un algorithme polynômial

La classe NP (NonDeterministic Polynomial time)

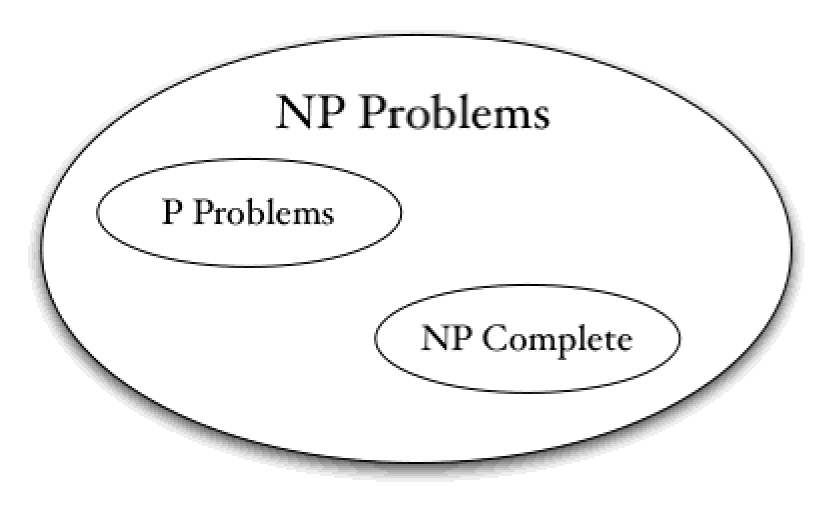
- L'ensemble des **problèmes de décision** pour lesquels on possède pour chaque « oui » (instance I) un schéma de vérification, vérifiant en temps polynômial la validité de la réponse « oui »

Problème NP-Complet

- Un **problème de décision** A dans NP est NP-complet si tous les autres problèmes de la classe NP se transforment polynomialement dans le problème A.

Classification des problèmes (2)

Relations entre P et NP



- $P \subseteq NP$
- Conjecture : $P \neq NP$
- 1 million de dollars à gagner !
- http://www.claymath.org/millenium-problems/

RECHERCHE OPERATIONNELLE

Définitions (1)

Cambridge Dictionary

- Operational research UK (US operations research): The systematic study of how best to solve problems in business and industry

Wikipedia

- Operations research, operational research, or simply OR, is the use of mathematical models, statistics and algorithms to aid in decision-making

Roadef (Société Française de RO-AD)

- Recherche Opérationnelle : approche scientifique pour la résolution de problèmes de gestion de systèmes complexes

Définitions (2)

Plus précisément

- Méthodes scientifiques pour résoudre des problèmes d'optimisation liés aux organisations du monde réel
- Une discipline à la croisée des mathématiques et de l'informatique :
  - Prolongement de l'algorithmique
  - Manipulation des structures plus élaborées : graphes, polyèdres, …
  - Domaine d'application de la théorie de la complexité algorithmique
- Une boîte à outils de méthodes, tant positives que négatives, pour aborder sainement et sereinement les problèmes d'optimisation

Secteurs d'application

Applications

- Industrie minière
- Transports : routier, ferroviaire, fluvial, aérien
- Gestion de la chaîne logistique
- Gestion de l'énergie
- Santé
- Planification de la production
- Problèmes d'ordonnancement
- Télécommunications
- Economie et finances
- Emplois du temps
- …

Entreprises & RO

Grands groupes français avec des pôles R&D en RO

- Airfrance
- SNCF
- EDF
- France Telecom
- Bouygues
- GDF Suez
- La Poste
- Renault
- Air Liquide
- SFR
- Google

Résolution d’un problème

- **Résolution des problèmes**
  - **Problèmes dans P**
    - **Algorithmes polynômiaux**
  - **Problèmes dans NP**
    - **Méthodes exactes**
    - **Méthodes approchées**

Face à un problème

Comprendre le problème posé :

- Contraintes, objectifs, simplifications

Modéliser le problème :

- Graphes, programmation linéaire, branch & bound (algorithme par séparation et évaluation), heuristiques, …

Connaitre les propriétés du modèle :

- Étude de complexité : que peut-on espérer pour le temps de résolution imparti ?

Résoudre le problème :

- Utiliser un solveur
- Mise au point d'algorithmes

Interpréter la solution :

- Interpréter la solution numérique obtenue en termes concrets

Quelques problèmes

Problèmes polynômiaux

- Programmation linéaire en continues
- Trouver le plus court chemin entre 2 sommets
- Trouver un chemin passant 1 fois par chaque arc (Eulérien)
- Maximiser un flot de véhicules dans un réseau de transport

Problèmes NP-complets

- Programmation linéaire en nombre entiers
- Problème du sac à dos
- Problèmes d'ordonnancement
- Problème du voyageur de commerce
- Trouver un chemin passant 1 fois par chaque sommet (Hamiltonien)
- Problème de la clique : Trouver k sommets connectés 2 à 2 par des arêtes
- Colorier les sommets d'un graphe

Contenu du cours

Contenu

- Graphes orientés
  - Définitions
  - Quelques exemples de graphes
- Graphes non-orientés
  - Définitions
  - Quelques exemples de graphes
- Fermeture transitive d’un graphe
- Graphe sans circuit
- Noyau d’un graphe
- Recherche d’un chemin de longueur min ou max
- Algorithmes de résolution :
  - Ford, Bellman-Kalaba, Djikstra
- Flot de coût minimum
- Problèmes d’ordonnancement

Organisation du cours

Volume horaire

- CM : 22 heures
- TD : 20 heures
- TP : mini-projet

- Contrôles continus